

CECS401  
Fundamentals of Spoken Language Processing

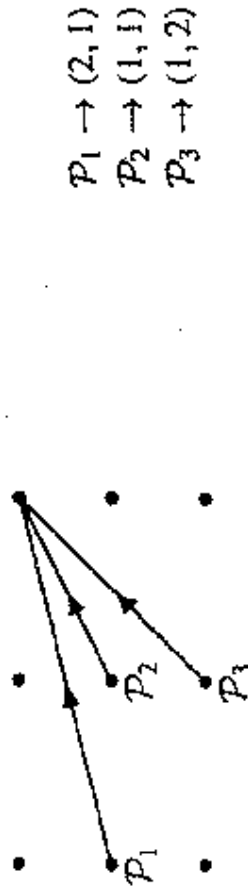
Note-14  
Thursday 10/14/99

## I. Speech Pattern Comparison using Dynamic Time Alignment

### Global constraint

The global range of time-warping paths can be derived from local continuity constraints or local path patterns and for reducing search space.

Example



Global range

## Slope weighting

Assign weight to different path patterns to encourage or discourage certain types of moves

Example

(a)  $m(k) = \min[\phi_x(k) - \phi_x(k-1), \phi_y(k) - \phi_y(k-1)]$



(b)  $m(k) = \max[\phi_x(k) - \phi_x(k-1), \phi_y(k) - \phi_y(k-1)]$



(c)  $m(k) = \phi_x(k) - \phi_x(k-1)$



(d)  $m(k) = \phi_x(k) - \phi_x(k-1) + \phi_y(k) - \phi_y(k-1)$



Figure 4.43 Type III local continuity constraints with four types of slope weighting (after Meyers et al. [23]).

## DTW Solutions

### Definitions

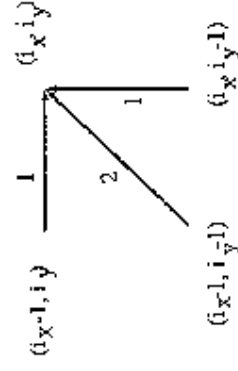
- $\phi = (\phi_x, \phi_y) \sim$  warping path
- $m(k) \sim$  slope weight
- $M_\phi = \sum_{k=1}^T m(k) \sim$  normalization factor
- $d(\phi_x(k), \phi_y(k)) = d(x_{\phi_x(k)}, y_{\phi_y(k)}) \sim$  local distance
- $D(i_x, i_y) \sim$  cumulative distance of the optimal path ending at  $(i_x, i_y)$

Distance between templates  $X$  and  $Y$ :

$$d(X, Y) = D(T_x, T_y) = \min_{(\phi_x, \phi_y)} \frac{1}{M_\phi} \sum_{k=1}^T d(\phi_x(k), \phi_y(k)) m(k)$$

## DTW recursion procedure

### Example



Initialization:  $D(1, 1) = 2d(1, 1)$

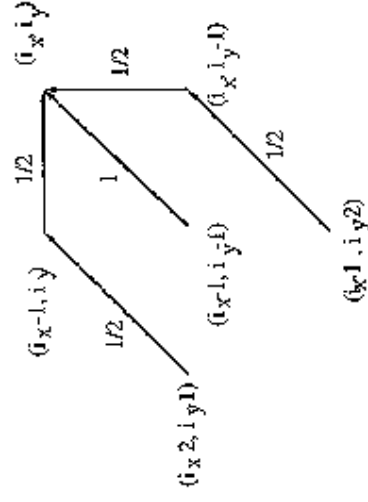
Recursion:

$$D(i_x, i_y) = \min \begin{cases} D(i_x - 1, i_y) + d(i_x, i_y) \\ D(i_x - 1, i_y - 1) + 2d(i_x, i_y) \\ D(i_x, i_y - 1) + d(i_x, i_y) \end{cases}$$

with  $(i_x, i_y)$  cover the global search range

Termination:  $D(X, Y) = \frac{1}{M_\phi} D(T_x, T_y)$

## Example



Initialization:  $D(1, 1) = d(1, 1)$

Recursion:

$$D(i_x, i_y) = \min \begin{cases} D(i_x - 2, i_y - 1) + \frac{1}{2}(d(i_x - 1, i_y) + d(i_x, i_y)) \\ D(i_x - 1, i_y - 1) + d(i_x, i_y) \\ D(i_x - 1, i_y - 2) + \frac{1}{2}(d(i_x, i_y - 1) + d(i_x, i_y)) \end{cases}$$

with  $(i_x, i_y)$  cover the global search range

Termination:  $D(X, Y) = \frac{1}{M_\phi} D(T_x, T_y)$

## **Multiple Template Training Using DTW**

Issue: how to derive representative reference templates from a set of training tokens of a given word?

Casual training:

Use each training token as a reference template, suitable for speaker-dependent tasks.

Clustering:

Derive a small set of representative reference templates from a large set of training tokens using a certain clustering criterion.

## **Modified K-means algorithm for template clustering**

For each given word, denote the set of training tokens as  $\Omega = \{X_1, X_2, \dots, X_L\}$ .

### Distance between tokens

$$d_{ij} = \frac{1}{2}[d(X_i, X_j) + d(X_j, X_i)]$$

### Clustering objective

Partition the training set  $\Omega$  into  $N$  disjoint clusters  $\Omega_i$ , i.e.,

$$\Omega = \bigcup_{i=1}^N \Omega_i, \text{ within each } \Omega_i \text{ the tokens are similar.}$$

### Cluster center

Minimax center:

Consider the cluster  $\Omega_k$ . For a token  $X_i \in \Omega_k$ , its largest distance to other tokens  $X_j \in \Omega_k$  is  $\max_{j \in \Omega_k} d_{ij}$ .

The minimax center of  $\Omega_k$  is defined as  $X_{i^*} \in \Omega_k$ , with

$$i^* = \arg \min_{i \in \Omega_k} \left\{ \max_{j \in \Omega_k} d_{ij} \right\}$$

Average center:

Use DTW to time-align all the tokens  $X_i \in \Omega_k$  to the minimax center  $X_{i^*} \in \Omega_k$

Average the time-normalized tokens to obtain the center  $\bar{X}^{(k)}$

The training procedure:

The procedure iterates to find an increasing number of clusters

$j = 1, 2, \dots, j_{max}$ .

For each fixed  $j$ , a K-means clustering is performed.

Define  $\omega_{j,i}^k$  as the  $i$ th cluster in the  $k$ th iteration when the total cluster is  $j$ , and  $y\left(\omega_{j,i}^k\right)$  as its cluster center.

- Step-5  
If  $j = j_{max}$ , stop, otherwise go to Step-6.
- Step-6 Cluster splitting  
Select the least compact cluster  $\omega_{j,i^*}^{k+1}$  and split it into two clusters

$$i^* = \arg \max_i \left( \Delta_{j,i}^{k+1} / \left| \omega_{j,i}^{k+1} \right| \right)$$

The split cluster centers are chosen as the most distant pair of tokens  $X_m$  and  $X_l$  in  $\omega_{j,i^*}^{k+1}$ , i.e.,

$$d_{m,l} > d_{p,q} \quad (m, l) \neq (p, q)$$

Set  $j = j + 1$ ,  $k = 1$ , go back to Step-2.

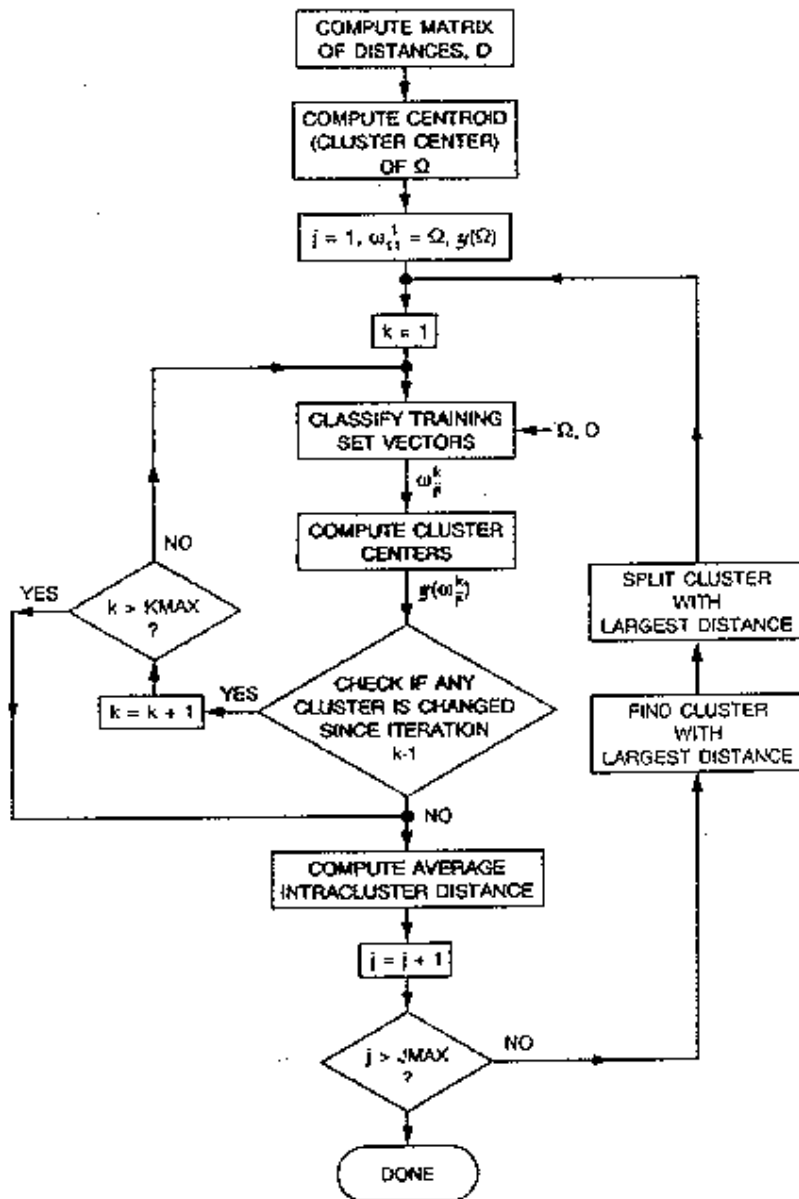


Figure 5.11 A flow diagram of the MKM clustering procedure (after Wilpon and Rabiner [7]).

### Recognition using multiple template

Use K-nearest neighbor rule:

For word  $m$ ,  $1 \leq m \leq M$ , sort the template distances as

$$d^{m(1)} \leq d^{m(2)} \leq \dots \leq d^{m(N)}$$

Compute the average distance as

$$d^m = \frac{1}{K} \sum_{k=1}^K d^{m(k)}$$

Classify test template X as the word  $m^*$  with

$$m^* = \arg \min_{1 \leq m \leq M} d^m$$

- Step-1 Initialization  
set  $j = 1, i = 1, k = 1, \omega_{1,1}^1$ .
- Step-2 Nearest neighbor assignment
  - Assign each token to a cluster  $\omega_{j,i}^k$ ; based on minimum token-center distance
  - Accumulate the intracluster distance  $\Delta_{j,i}^k$
- Step-3 Update cluster centers  $y(\omega_{j,i}^{k+1}), 1 \leq i \leq j$
- Step-4 K-means convergence check
  - If
    - $y(\omega_{j,i}^{k+1}) = y(\omega_{j,i}^k), 1 \leq i \leq j$ , or
    - $k = k_{max}$ , or
    - $\left( \sum_{i=1}^j \Delta_{j,i}^k - \sum_{i=1}^j \Delta_{j,i}^{k+1} \right) / \sum_{i=1}^j \Delta_{j,i}^k < threshold$
  - then go to Step-5,
  - otherwise set  $k = k + 1$  and go back to step-2.